Amendments to the Claims:

Please cancel Claims 18, 27, and 35; amend Claims 17, 31, 33, and 34; and add Claim 36 as indicated in the following listing of claims, which replaces all prior versions and listings, of claims in the application.

Listing of Claims:

1. – 16. (Canceled)

17. (Currently Amended) A method of processing a transport steam comprising the steps of:

(a) parsing the transport stream to derive multiple elemental streams including associated program identifiers;

(b) using the associated program identifiers to assign each stream a direct memory access channel;

(c) associating each direct memory access channel with a specific location in the memory of a host computer by providing at least one data descriptor comprising a frame descriptor and at least one data descriptor comprising a channel context descriptor, wherein the frame descriptor associates a region in local memory that is the source of data, and the channel context descriptor associates a region in the host memory where data is to be stored; and

(d) performing direct memory access transfers of the multiple elementary streams to corresponding locations in the memory of the host computer using the direct memory access channels,

wherein the multiple elemental streams are transferred between a local memory and the memory of the host computer and at least two sets of data descriptors are provided, wherein the first set comprises a local memory frame descriptor and a local memory channel context descriptor, and the second set comprises a host memory frame descriptor and a host memory channel context descriptor.

18. (Canceled).

19. (Previously Presented) The method of claim 17 wherein the multiple elemental streams are transferred between a transport controller and the memory of the host computer.

20. (Canceled).

21. (Previously Presented) The method of claim 19 wherein the direct memory access transfer is an automatic programmable transport interface operation wherein data is not buffered in a local memory prior to the transfer to the memory of the host computer.

22. – 23. (Canceled).

24. (Previously Presented) The method of claim 17 further comprising the step of transferring the multiple elementary streams to an end user system.

25. (Previously Presented) The method of claim 24 wherein the end user system comprises an audio-visual system and wherein the step of transferring the multiple elementary steams to an end user system comprises transferring the multiple elementary streams through an audio-visual interface.

26. (Previously Presented) The method of claim 24 wherein the end user system comprises a networked computer system and the step of transferring the multiple elementary streams to an end user system comprises transferring the multiple elementary streams through a network interface.

27. – 30. (Canceled).

31. (Currently Amended) The system of claim ~~27~~ 33 wherein the transport controller is configured by the local memory to associate the program identifiers with corresponding DMA channels so that data is directly transferred between the transport controller and the host memory without being buffered in the local memory prior to transfer.

32. (Previously Presented) The method of claim 17 wherein each of the frame descriptors include the following fields: (1) a segment N pointer field; (2) a segment N size field, which identifies the size of the Nth segment; (3) an end-of-field descriptor bit to identify the last segment in a frame; (4) a last segment number field, which is updated at the end of a data transfer to identify the last filled segment in the frame used for that transfer; and (5) a remaining bytes field, which is updated at the end of a data transfer to identify the number of bytes remaining in the last segment used for that transfer.

33. (Currently Amended) ~~The method of claim 17~~ A system for receiving and processing a transport stream comprising:

a receiver for the transport stream having a local memory and a transport controller; and

a host computer having a host memory, a host central processing unit (CPU) and a direct memory access (DMA) engine;

wherein the transport controller is configured to parse the transport stream to derive multiple elemental streams including associated program identifiers and

wherein the local memory is configured to assign each stream a DMA channel using the associated program identifiers, and associate each DMA channel with a specific location in the host memory by providing at least one data descriptor comprising a frame descriptor and at least one data descriptors comprising a channel context descriptor, wherein the

frame descriptor associates a region in local memory that is the source of the data, and the channel context descriptor associates a region in the host memory where data is to be stored ; and

wherein the DMA engine uses the DMA channels to transfer the multiple elementary streams to corresponding locations in the host memory,

wherein each of the channel context descriptors include the following fields: (1) a channel N frame descriptor pointer field, which uniquely associates DMA channel N with a particular block of data; (2) a current address field, which points to the current host or local memory byte address as the data transfer progresses; (3) a segment remaining-bytes field, which identifies the number of bytes remaining in the current field descriptor segment and is updated in real time as the data transfer progresses; (4) a segment number, which identifies in real time the current segment in the frame used for the transfer; (5) an end-of-frame descriptor bit, copied from the end-of-frame descriptor bit of the frame descriptor to identify whether the current segment is the last in a frame; and (6) a channel control byte field.

34. (Currently Amended) ~~The system of claim 27~~ A system for receiving and processing a transport stream comprising:

a receiver for the transport stream having a local memory and a transport controller; and

a host computer having a host memory, a host central processing unit (CPU) and a direct memory access (DMA) engine;

wherein the transport controller is configured to parse the transport stream to derive multiple elemental streams including associated program identifiers and

wherein the local memory is configured to assign each stream a DMA channel using the associated program identifiers, and associate each DMA channel with a specific location in the host memory by providing at least one data descriptor comprising a frame descriptor and at least one data descriptors comprising a channel context descriptor, wherein the frame descriptor associates a region in local memory that is the source of the data, and the channel context descriptor associates a region in the host memory where data is to be stored ; and

wherein the DMA engine uses the DMA channels to transfer the multiple elementary streams to corresponding locations in the host memory,

wherein each of the frame descriptors include the following fields: (1) a segment N pointer field; (2) a segment N size field, which identifies the size of the Nth segment; (3) an end-of-field descriptor bit to identify the last segment in a frame; (4) a last segment number field, which is updated at the end of a data transfer to identify the last filled segment in the frame used for that transfer; and (5) a remaining bytes field, which is updated at the end of a data transfer to identify the number of bytes remaining in the last segment used for that transfer.

35. (Canceled).

36. (New) The system of claim 34 wherein the transport controller is configured by the local memory to associate the program identifiers with corresponding DMA channels so that data is directly transferred between the transport controller and the host memory without being buffered in the local memory prior to transfer.